A SINGLE NEURON MODEL FOR SOLVING BOTH PRIMAL AND DUAL LINEAR PROGRAMMING PROBLEMS

P. Pandian [#1], G. Selvaraj [#2]

[#] Dept. of Mathematics, School of Advanced Sciences, VIT University
Vellore, Tamilnadu, India-632014
[1] pandian61@rediffmail.com
[2] gselvaraj89@gmail.com

Abstract— A new neural network with a single neuron for solving both a linear programming problem and its dual is presented in this paper. Based on the duality gap problem, the network and its energy function are defined. The proposed neural network based on a nonlinear dynamical system uses only simple hardware in which no analog multipliers are required and is proved to be globally asymptotically stable to the exact solution. Some simulation results are presented for showing the efficiency and simplicity of the proposed neural network.

Keyword-Linear programming problem, Duality gap problem, Energy function, Neural network.

I. INTRODUCTION

An optimization of a linear function that satisfies a set of linear equality and/or inequality constraints is known as a linear programming (LP) problem. LP problem has many applications in Management Science, Science, Engineering, Medical Science and Technology. In real life situations, LP models with a large size is considered to be one of the basic problems widely encountered. Solving such LP problems within a time of the order of 100 ns, using the traditional algorithms of digital computer such as the Simplex method [1] and Karmarkar's algorithm [2] cannot do this point, or its use is very expensive. The employing neural networks is one possible approach to solve such problems. Neural networks or Artificial neural networks are dynamic systems that consist of highly interconnected and parallel nonlinear processing elements called artificial neurons that show extreme efficiency in computation. The neural network algorithms have many computational advantages over the traditional algorithms. The most important advantages of the neural networks are massively parallel processing and fast convergence.

In the literature, many researchers have applied various types of artificial neural networks to solve several classes of constrained optimization problems efficiently. Solving the LP problem using neural network was first proposed by Hopfield and Tank [3]. They used a LP circuit in their net. Followed by them many researchers have worked on neural network implementation for LP problems which transform the given problem into dynamical systems. Kennedy and Chua [4] proposed an improved neural network model of Hopfield and Tank's model which is always guaranteed convergence, but it converges to only an approximation of the optimal solution. Maa and Shanblatt [5] introduced a two-phase neural network model which converges to the exact solution, but it is relatively complex and still requires some parameter tuning. Xia [6] proposed a network for solving LP problems in which the energy function requires no parameter tuning. Xia's network solves both primal and dual problems simultaneously with the help of two layers of neurons which was known by two system of ordinary differential equations. Nguyen [7] used a recurrent neural network based on a nonlinear dynamical system having interconnected two layers of neurons for solving LP problems which is simple and more intuitive.

Malek and Yari [8] proposed a neural network for solving LP problems and its dual which has its energy function as Lyapunov function. Malek and Alipour [9] constructed a recurrent neural network with no parameter setting to solve LP problems and quadratic programming problems. Ghasabi-Oskoei et al. [10] developed a new neural network model based on a nonlinear dynamical system, using arbitrary initial conditions. It converges very fast to the exact primal and dual solutions simultaneously. Gao and Liao [11] constructed a new neural network for solving linear and quadratic programming problems in real time by introducing some new vectors. Cichocki et al. [12] proposed a neural network model without duality concepts for solving an LP problem which is transformed into a system of differential equations whose equilibrium point is the solution of the LP problem.

In this paper, we present a new neural network based on duality gap concept and on a nonlinear dynamic system for solving both primal and dual LP problems. It is presented in which only one neuron is used for computation, that is, only one set of ordinary differential equations are used. First we construct the duality gap problem for the given LP problem. Then, using the proposed neural network, we transform the duality gap problem into a system of differential equations in such a way that the equilibrium point of the dynamic system is the solution to the given LP problem and its dual. In the proposed network, only simple hardware without analog multipliers is used. We derive that the network is globally asymptotically stable to the exact solution and show the efficiency and simplicity of the proposed neural networks with the help of some simulation results. The proposed network model can help decision makers to come across a situation in real life to analyze the solutions of both primal and dual LP models.

II. PRELIMINARIES

In this section, some definitions and results in linear programming [1] and neural network [13] are given which will useful for the developing the proposed neural network model.

Consider the following linear programming problem

(P) Maximize $C^T X$

subject to $AX \le B,\ X \ge 0.$

where $C^T = (c_1, c_2, ..., c_n) \in R^n$, $X^T = (x_1, x_2, ..., x_n) \in R^n$, $B^T = (b_1, b_2, ..., b_m) \in R^m$ and A is an mxn matrix over *R*, a field of real numbers.

Now, the dual to the problem (P), (D) is given below

(D) Minimize $B^T Y$

subject to $A^T Y \ge C,\ Y \ge 0.$

where the vector $Y^T = (y_1, y_2, ..., y_m) \in R^m$ is called the dual vector.

Remark 2.1: (a) The problem (P) contains '$\ge$' type constraints, convert the constraints into '$\le$' type constraints and then find its dual.

(b) The problem (P) contains '$=$' type constraints, add unrestricted conditions to their dual variables in the problem (D).

(c) The problem (P) contains '$\ge$' and '$=$' type constraints, use both Remark 2.1 and 2.2 for finding the dual problem (D).

Result 2.1: (a) Suppose that X and Y are feasible solutions to (P) and (D) respectively. Then, $C^T X \le B^T Y$.

(b) Suppose that X^0 and Y^0 are feasible solutions to (P) and (D) respectively. If $C^T X^0 = B^T Y^0$, then X^0 and Y^0 are optimal solutions to their respective problems.

(c) If the problem (P) has an optimal solution, then so does the dual, and the optimal values of their respective objective functions are equal.

Definition 2.1: A function $P : R^n \to R$ is called a penalty function for the constrained optimization problem (P) if it satisfies the following conditions:

(i) P is continuous and positive and

(ii) $P(X) = 0$ if and only if X is feasible for (P).

Theorem 2.1: (Lyapunov function method) Let x^* be an equilibrium point for the system $\frac{dx(t)}{dt} = f(x(t))$.

Let $V : R^n \to R$ be a continuously differentiable function such that

$V(x^*) = 0$ and $V(x) > 0,\ \forall\, x \ne x^*$;

$V(x) \to \infty$ *when* $\|x\| \to \infty$ and

$\frac{dV(x)}{dt} < 0 \quad \textit{for all } x \ne x^*$.

Then, $x = x^*$ globally asymptotically stable.

Definition 2.2: Any scalar function $V(x)$ that satisfies the requirements of the Theorem 2.1 is called a Lyapunov function for the equilibrium state $x = x^*$.

III. DUALITY GAP PROBLEM

Consider the following LP problem related to (P), called the duality gap problem for the problem (P), (G):

(G) Minimize $B^T Y - C^T X$

subject to

$AX \le B$, $A^T Y \ge C$, $X \ge 0$, $Y \ge 0$.

Now, we derive the following theorems which help us to understand the relation between the problems (P) and (D) and the problem (G) and also, used in the proposed method.

Theorem 3.1: If X and Y are optimal solutions to (P) and (D) respectively, then {X, Y} is an optimal solution to the problem (G).

Proof: From the Result 2.1(b) & (c), it follows.

Theorem 3.2: If {X, Y} is an optimal solution to the problem (G), then X and Y are optimal solutions to the problems (P) and (D) respectively.

Proof: Clearly, X and Y are feasible solutions to (P) and (D) respectively.

Suppose that X is not optimal solution of (P).

Then, there exists a feasible solution, U to the problem (P) such that $C^T U > C^T X$.

Now, since {U, Y} is a feasible solution to the problem (G), we have

$$B^T Y - C^T X \le B^T Y - C^T U .$$

This implies that $C^T U \le C^T X$ which contradicts the result $C^T U > C^T X$.

Therefore, X is an optimal solution of (P).

Suppose that Y is not optimal solution of (D).

Then, there exists a feasible solution, V to the problem (D) such that $B^T V < B^T Y$.

Now, since {X, V} is a feasible solution to the problem (G), we have

$$B^T Y - C^T X \le B^T V - C^T X .$$

This implies that $B^T V \ge B^T Y$ which contradicts the result $B^T V < B^T Y$.

Therefore, Y is an optimal solution of (D).

Hence the theorem is proved.

Remark 3.1: If (P) is minimization type, then the objective function of (G) becomes $C^T X - B^T Y$.

IV. THE PROPOSED NEURAL NETWORK MODEL

Now, the LP problem (G) after converting all inequality constraints into equality constrains by adding slack variables or surplus variables can be written as follows:

(G) Minimize $D^T Z$

subject to

$HZ = F$, $Z \ge 0$

where $Z = \begin{pmatrix} X & Y & S \end{pmatrix}^T$, $H = \begin{pmatrix} A & 0 & \mathrm{I_m} \\ 0 & A^T & -I_n \end{pmatrix}$, $F = \begin{pmatrix} B \\ C \end{pmatrix}$, $D^T = \begin{pmatrix} -C^T & B^T & 0_s \end{pmatrix}$

and $S = (s_1, s_2, ..., s_m, t_1, t_2, ..., t_n)^T$, $s_i, i = 1,2,...,m$, are slack variables and $t_j, j = 1,2,...,n$, are surplus variables.

Now, the energy function represents the behavior of the networks and supports the direction to search out solutions for real-time optimization. For solving the LP problem, the energy function can be defined in any one of the three different methods [14] namely, Penalty method, Lagrange multipliers method and Primal dual method such that the energy function has its minimum at the solution of the given LP problem. Here, we construct an energy function, $E(z)$ for the proposed neural network for solving the problem (G) with the help of penalty function as follows:

$$E(Z) = \frac{1}{2}\left\| D^T Z \right\|^2 + kP(e)$$

where $P(e) = \frac{1}{2}e^2$, $e = HZ - F$, $k > 0$ is the penalty parameter.

Note that from the properties of convex functions [15], we can conclude that $E(Z)$ is convex.

We now, develop a neural network model for solving the duality gap problem (G) for the given LP problem (P) using the newly defined energy function $E(Z)$.

Now, since the energy function $E(Z)$ attains its minimum at an optimal solution to (G) and $E(Z)$ is convex [14], we have to find the local minimum of the energy function. For finding local minimum of the energy function by the gradient search approach, we get the neural network model as given below.

$$\frac{dZ}{dt} = -\nabla E(Z)$$

That is,

$$\frac{dZ}{dt} = -[D(D^T Z) + kP'(e)]$$

Since $P'(e) = \frac{\partial P}{\partial Z} = \frac{dP}{de} . \frac{\partial e}{\partial z} = H^T e \; = H^T (HZ - F)$, it follows that

$$\frac{dz}{dt} = -[D(D^T Z) + kH^T (HZ - F)] \qquad (1)$$

Now, the neural network representation of (1) is shown below

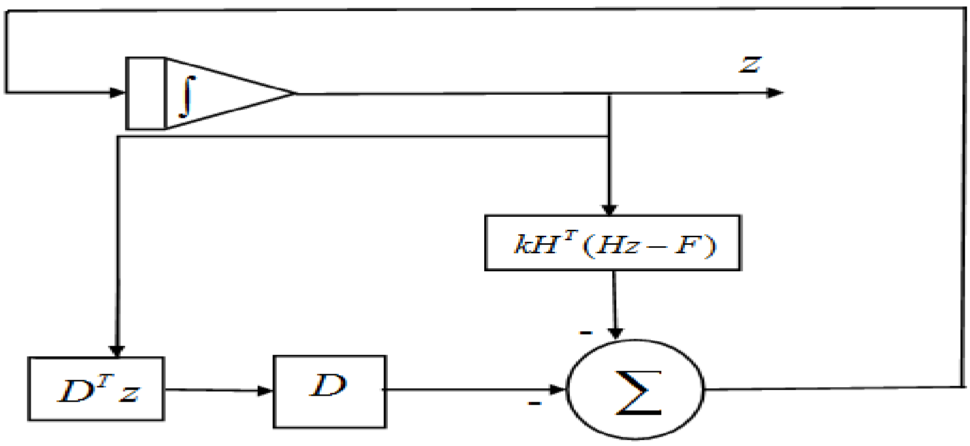

Fig.1 Network representation

For solving the system (1) of the differential equations, we can obtain the values of z which give the solutions of the problems (P) and (D).

Now, we derive the stability of the proposed neural network.

Theorem 4.1: $E(Z)$ is a Lyapunov function for the dynamic system (1) and the dynamic system (1) is globally asymptotically stable at the equilibrium point.

Proof: Let $Z = Z^*$ be an equilibrium point of the system (1).

Now, since $\|D^T Z\|^2 > 0 \; and \; P(Z) > 0 , \; \forall \, Z \neq Z^*$, we have

$$E(Z^*) = \frac{1}{2}\|D^T Z^*\|^2 + kP(Z^*) = 0 \quad \text{and} \quad E(Z^*) > 0 .$$

Now, $\frac{dE(Z(t))}{dt} = \frac{\partial E(Z)}{\partial Z} \cdot \frac{dZ}{dt}$

$$= [D(D^T Z) + kH^T (HZ - F)] \cdot (-[D(D^T Z) + kH^T (HZ - F)]) \quad \text{since by (1)}$$

$$= -[D(D^T Z) + kH^T (HZ - F)]^2$$

$$< 0 \qquad \forall \, Z \neq Z^*$$

Therefore, $E(Z)$ is a Lyapunov function for the system (1) and this yields the globally asymptotic stability at the equilibrium $Z = Z^*$ in the sense of Lyapunov.

Hence the theorem is proved.

Now, since $E(Z)$ is Lyapunov function at an optimal point of the problem (G), the dynamic system (1) governed by the energy function $E(z)$ converges to the optimal solution to the LP problem (G). This implies that, the dynamic system converges to the optimal solutions of the problems (P) and (D).

Remark 4.1: For solving both the problems (P) and (D) by a network, we are enough to solve the problem (G) by the proposed neural network.

V. Simulation Examples

In order to show the efficiency and simplicity of the proposed neural network, we present some simulation results. All the computer simulation results presented here have been obtained by using built-in function ode45 (which is equivalent to Runge-Kutta method) in MATLAB program.

Example 5.1: Consider the following LP problem

Minimize $-x_2 + 6$

subject to

$$2x_1 - x_3 = -5$$

$$x_2 + x_4 = 5$$

$$x_i \geq 0, i = 1, 2, 3, 4$$

Now, using zero as initial state for all decision variables, we obtain following results by the proposed neural network model:

$X = (x_1, x_2, x_3, x_4)$ = (0.00000 , 4.99979 , 2.50000 , 0.00020);

$Y = (y_1, y_2)$ = (0.00001, -0.999997) and Dual gap, G = 0.000245.

The convergence to the optimum solution to the problem is shown below (Fig.-2.)

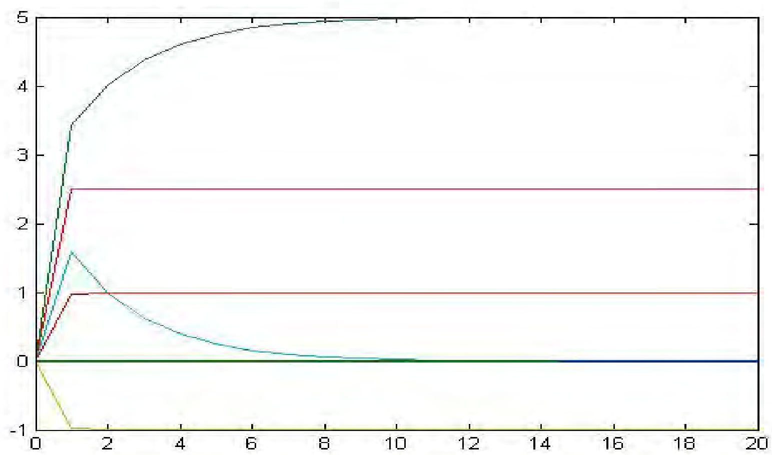

Fig-2. Convergence of solutions

The following table gives the comparative study of the proposed method with other methods:

TABLE I
Comparison with other methods

Method	Primal	Dual
Our proposed network	(0.00000 , 4.99979 , 2.50000 , 0.00020)	(0.00001, -0.999997)
Xia's network	(0, 5.00081, 2.50001, 0)	-
Ngyuen's network	(1.84255, 4.77529666, 8.7760854233, 0),	(0.03212, -1.56997)
Simplex method	(0, 5, 2.5, 0)	(0, -1)

Example 5.2: Consider the following LP problem

Maximize $x_1 + 9x_2 + x_3$

subject to

$$x_1 + 2x_2 + 3x_3 \le 9$$
$$3x_1 + 2x_2 + 2x_3 \le 15$$
$$x_i \ge 0, \ \forall\, i = 1,2,3$$

Now, using zero as the initial state for all decision variables, we obtain the following solutions by the proposed neural network model :

$X = (x_1, x_2, x_3) = (0,\ 4.5,\ 0)$, $Y = (y_1, y_2) = (4.5,\ 0)$ and Dual gap, G= 0.

The convergence of the solution of the problem is shown below (Fig-3.):

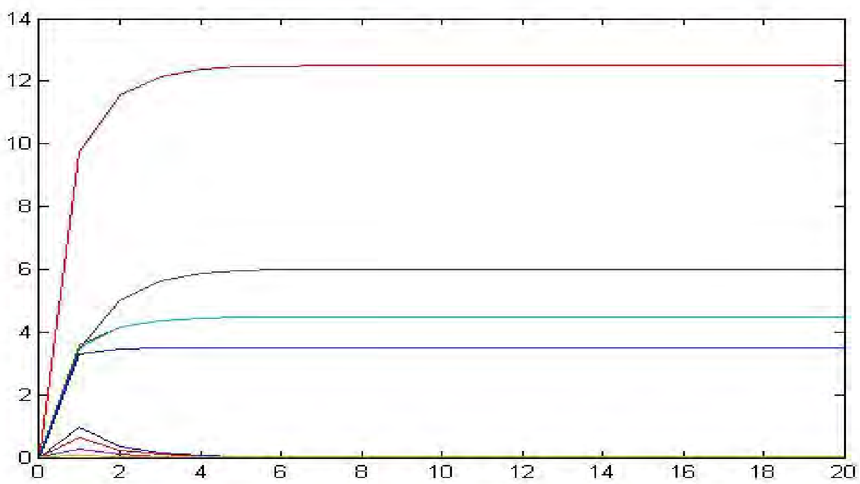

Fig-3. Convergence of solutions

The following table gives the comparative study of the proposed method with other methods:

TABLE II
Comparison with other methods

Method	Primal	Dual
Our proposed networks	(0, 4.5, 0)	(4.5, 0)
Ngyuen's Network	(0, 4.4388035, 0)	(4.622766, 0)
Simplex method	(0, 4.5, 0)	(4.5, 0)

For solving non-co-operative two-person zero-sum game problems by LP technique, we have to find optimal solutions to both primal and dual problems. So, the proposed network is very much helpful to solve such problems which is illustrated by the following example.

Example 5.3: Consider the following non-cooperative (competitive) two-person zero-sum game

$$A = \begin{pmatrix} 180 & 156 & 90 \\ 90 & 180 & 156 \\ 180 & 156 & 177 \end{pmatrix}$$

Now, let the optimal strategies for the players A and B be

$$S_A = \begin{pmatrix} A_1 & A_2 & A_3 \\ p_1 & p_2 & p_3 \end{pmatrix} \text{ and } S_B = \begin{pmatrix} B_1 & B_2 & B_3 \\ q_1 & q_2 & q_3 \end{pmatrix}$$

and the optimal value of the game be v.

Now, the duality gap problem corresponding to the Player A, (G) is given below:

Minimize G = $(x_1 + x_2 + x_3) - (y_1 + y_2 + y_3)$

subject to

$$180x_1 + 90x_2 + 180x_3 \geq 1; \qquad 156x_1 + 180x_2 + 156x_3 \geq 1;$$
$$90x_1 + 156x_2 + 177x_3 \geq 1; \qquad 180y_1 + 156y_2 + 90y_3 \leq 1;$$
$$90y_1 + 180y_2 + 156y_3 \leq 1; \quad 180y_1 + 156y_2 + 177y_3 \leq 1;$$
$$x_i \geq 0, \mathrm{i} = 1,2,3 \quad \text{and} \quad y_j \geq 0, \mathrm{j} = 1,2,3$$

where $x_i = \frac{p_i}{v}$, $\mathrm{i} = 1,2,3$ and $y_j = \frac{q_j}{v}$, $\mathrm{j} = 1,2,3$.

Now, the optimal solution to the duality gap problem by the proposed neural network model are given below:
X= (0.00088302, 0.00136406, 0.00399778) , Y=(0.00126079, 0.00473017, 0.00019947)
Duality gap, G = 0.00005443 rounding off for up to 4 decimals we have 0 as the dual gap and the optimum be 0.0062
Now, the optimal strategies for the players A and B are

$$S_A = \begin{pmatrix} A_1 & A_2 & A_3 \\ 0.1424 & 0.22 & 0.6448 \end{pmatrix}, \; S_B = \begin{pmatrix} B_1 & B_2 & B_3 \\ 0.20335 & 0.7629 & 0.03217 \end{pmatrix}$$

and the optimal value of the game is $v = 161.29$
The convergence to the optimum solution to the problem is shown below (Fig.-4.):

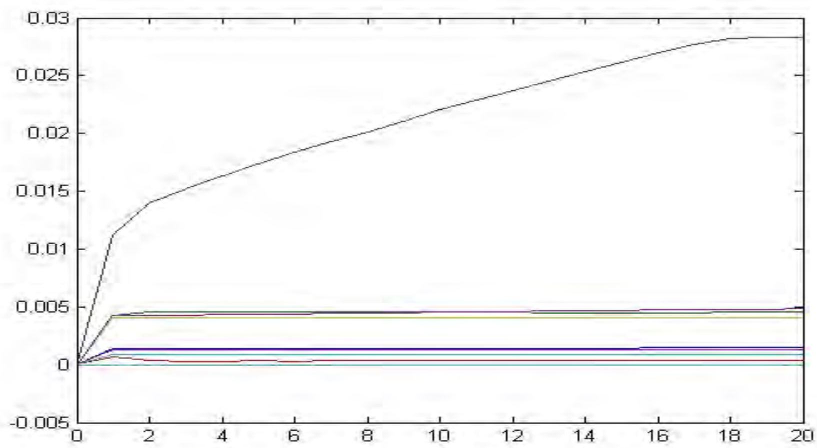

Fig-4.Convergence of solutions

The following table gives the comparative study with other networks:

TABLE III
Comparison with other methods

Method	Primal	Dual
Our proposed network	(0.00088302,0.00136406,0.00399778)	(0.00126079,0.00473017,0.00019947)
Ngyuen's network	(0,0,0)	(2.6711, 2.6555, 3.6244)
Simplex method	(0.00082, 0.00131, 0.00408)	(0.00131, 0.0049, 0)

VI. CONCLUSION

In this paper, we present a new simple neural network for solving both an LP problem and its dual. In the construction of new network, we use duality gap problem for the given LP problem and develop a new energy function such that its equilibrium point is an optimal solutions to both the LP problem and its dual. The dynamic behavior and performance of the proposed neural network have been illustrated through extensive computer simulations. An interesting and important feature of the proposed network is to obtain solutions for both LP problem and its dual by only one set of dynamic system. For solving real-time and large-scale LP problems with their duals simultaneously, the proposed neural network can serve as an effective computational model. In near future, we have a plan to extend the new neural network to fuzzy LP problems.

VII. References

[1] Edwin KP Chong, Stanislaw H. Zak. An Introduction to Optimization, John Wiley & Sons Inc , New York, 2001.

[2] N. Karmarkar, "A new polynomial-time algorithm for linear programming", Combinatorica, Vol. 4, pp. 373-395, 1984.

[3] D. W. Tank, J.J. Hopfield, "Simple neural optimization networks: An A / D converter, signal decision circuit and a linear programming circuit", IEEE Transactions on circuits and systems (CAS). Vol.33, pp. 533-541, 1986.

[4] M. P. Kennedy, L. O. Chua, "Neural networks for nonlinear programming", IEEE Transactions on Circuits and Systems. Vol.35, pp.554-562, 1988.

[5] C. Maa, M. Shanbaltt, "A two-phase optimization neural network", IEEE Trans. On Neural Networks, Vol 3, pp. 1003-1009, 1992.

[6] Y. Xia, "A new neural network for solving linear programming problems and its application", IEEE Transactions on Neural Networks, Vol. 7, pp. 525-529, 1996.

[7] K. V. Nguyen, "A nonlinear neural network for solving linear programming problems", in Proceedings of the 17th International Symposium on Mathematical Programming, Atlanta, Georgia, 2000.

[8] A. Malek, A. Yari, "Primal–dual solution for the linear programming problems using neural networks", Applied Mathematics and Computation, Vol. 167, pp. 198–211, 2005.

[9] A. Malek, M. Alipour, "Numerical solution for linear and quadratic programming problems using a recurrent neural network", Applied Mathematics and Computation, Vol. 192, pp. 27–39, 2007.

[10] H. Ghasabi-Oskoei, A. Malek, A. Ahmadi, "Novel artificial neural network with simulation aspects for solving linear and quadratic programming problems", Computers and Mathematics with Applications, Vol. 53, pp. 1439–1454, 2007.

[11] X. Gao, L. Z. Liao, "A new one-layer neural network for linear and quadratic Programming", IEEE Transactions on Neural Networks, Vol. 21, pp. 918-929, 2010.

[12] A. Cichocki, R. Unbehauen, K. Weinzierl, R. Htlzel, "A new neural network for solving linear programming problems", European Journal of Operational Research. Vol. 93, pp. 244-256, 1996.

[13] Madan M Gupta, Liang Jin, Noriyasu Homa. Static and Dynamical Neural Networks: From Fundamentals to Advanced Theory, John Wiley & Sons Inc., New Jersey, 2003.

[14] Ue-Pyng Wen, Kuen-Ming Lan, Hsu-Shih Shi., "A review of Hopfield neural networks for solving mathematical programming problems", European Journal of Operation Research, Vol. 198, pp. 675-687, 2008.

[15] Boyd S, Vandenberghe L. Convex Optimization, Cambridge University Press, New York, 2004.